

Amendments to the Claims

This listing of claims replaces all prior versions and listings of claims in the application.

Listing of Claims:

1. (Previously presented) A method of processing messages comprising:
receiving a markup language document describing a message, the markup language document being compliant with message rules associated with an online transaction processing (OLTP) language;
receiving a template for code in a program language, the template describing a data format for the message in a target program language;
automatically generating source code in the program language from the markup language document and the template, the automatically-generated source code depending on both the markup language document and the template, and configured to transform a payload message directly between the target program language data format and an online transaction processing (OLTP) language message format when compiled and executed;
compiling the automatically-generated source code;
using the compiled automatically-generated source code to transform the payload message directly between the target program language data format and the online transaction processing (OLTP) language message format.
2. (Canceled).
3. (Previously presented) The method of claim 1, wherein the message rules define markup tags for describing components of the message.
4. (Original) The method of claim 3, wherein the message is a request message from an application to an OLTP host.
5. (Original) The method of claim 3, wherein the message is a response message from an OLTP host to an application.

6. (Original) The method of claim 3, wherein the markup language is an extensible markup language.
7. (Previously presented) The method of claim 1, wherein the template comprises stylesheet data.
8. (Previously presented) The method of claim 7, further including using a stylesheet parser to automatically generate the source code.
9. (Previously presented) The method of claim 1, further including receiving a plurality of markup language documents for a plurality of message types and generating source code to convert messages of the plurality of message types between the OLTP language and the target language.
10. (Previously presented) The method of claim 1, wherein the target language is an object-oriented program language.
11. (Original) The method of claim 10, further including:
 - receiving one or more name-value pairs for the message according to the object-oriented program language;
 - generating a byte array according to the OLTP language; and
 - transmitting the byte array to an application.
12. (Original) The method of claim 10, further including:
 - receiving a byte array for the message according to the OLTP language; generating one or more name-value pairs according to the object-oriented program language; and
 - transmitting the name-value pairs to an application.
13. (Original) The method of claim 10, wherein the object-oriented program language is a Java language.
14. (Original) The method of claim 1, wherein the OLTP language is a gaming system language.

15. (Previously presented) A method of generating source code, comprising:
receiving a markup language document, the markup language document being compliant with message rules associated with an online transaction processing (OLTP) language;
receiving a template for code in a program language, the template describing a data format in a target program language for a message; and
automatically generating a source code based on the markup language document and the template, the source code depending on both the markup language document and the template, and enabling transformation of a payload message directly between the target program language data format and an OLTP language message format.
16. (Original) The method of claim 15, wherein the message rules define markup tags for describing components of the message.
17. (Original) The method of claim 16, wherein the message is a request message from an application to an OLTP host.
18. (Original) The method of claim 16, wherein the message is a response message from an OLTP host to an application.
19. (Original) The method of claim 16, wherein the markup language is an extensible markup language.
20. (Previously presented) The method of claim 15, wherein the template comprises stylesheet data.
21. (Previously presented) The method of claim 20, further including using a stylesheet parser to automatically generate the source.
22. (Previously presented) The method of claim 15, further including receiving a plurality of markup language documents for a plurality of message types and generating source code to convert the plurality of message types between the OLTP language and the target language.

23. (Currently amended) A method of processing messages comprising:
- receiving one or more extensible markup language (XML) documents, the XML documents being compliant with message rules associated with an online transaction processing (OLTP) language, the OLTP language being a gaming system language and the message rules defining markup tags for describing components of the a message;
 - automatically compiling the XML documents into source code in an object-oriented program language based on stylesheet data, the stylesheet describing a data format for the message in a target program language, the source code configured, when compiled and executed, to transform a message directly between the target program language data format and an OLTP language message format;
 - compiling the source code;
 - using the compiled source code to transform a payload message directly between the target language data format and the OLTP language message format; and
 - repeating the receiving and compiling of XML documents and the compiling of source code for a plurality of message types.
24. (Original) The method of claim 23, further including using a stylesheet parser to compile the XML documents.
25. (Original) The method of claim 23, further including: receiving one or more name-value pairs for the message according to the program language; generating a byte array according to the OLTP language; and transmitting the byte array to an application.
26. (Original) The method of claim 23, further including: receiving a byte array for the message according to the OLTP language; generating one or more name-value pairs according to the program language; and transmitting the name-value pairs to an application.
27. (Original) The method of claim 23, wherein the object-oriented program language is a Java language.
28. (Previously presented) A message processing apparatus, comprising: an application server including a development module stored on the application server, the development

module configured to receive a markup language document and a template for code in a program language, the template describing a message data format in a target program language, and configured to automatically generate a source code in the program language, the source code depending on both the markup language document and the template, the markup language document to be compliant with message rules associated with the OLTP language, and the source code configured to enable direct transformation of a message between the target program language message data format and an OLTP language message format;

a program compiler stored on the application server, the program compiler configured to compile the source code; and

a message parser stored on the application server, the message parser configured to transform a payload message directly between the target program language message data format and the online transaction processing (OLTP) language message format using the compiled source code.

29. (Canceled).

30. (Previously presented) The system of claim 28, wherein the message rules define markup tags for describing components of the message.

31. (Previously presented) The system of claim 28, wherein the development module further includes a stylesheet parser, the development module to use the stylesheet parser to automatically generate the source code based on stylesheet data.

32. (Currently amended) A machine readable medium comprising a set of stored instructions capable of being executed by a processor to:

receive a markup language document, the markup language document being compliant with message rules associated with an online transaction processing (OLTP) language;

receive a template for code in a program language, the template describing a data format in a target program language relating to a message; and

automatically generate a source code based on the markup language document and the template, the source code depending on both the markup language document and the template,

and configured to transform a payload message directly between a the target program language data format and an OLTP language message format, when the source code is compiled and executed.

33. (Original) The medium of claim 32, wherein the message rules define markup tags for describing components of the message.

34. (Previously presented) The medium of claim 32, wherein the template comprises stylesheet data.

35. (Previously presented) The medium of claim 32, wherein the instructions are further capable of being executed to receive a plurality of markup language documents for a plurality of message types and to generate source code for converting the plurality of message types between the OLTP language message format and the target language message format.

36. (Previously presented) The method of claim 1, further comprising:
providing an online gaming system by an OTLP host which uses the OLTP language, wherein the online gaming system processes wagering-related transactions; and
providing an application at a consumer location which uses the target program language, wherein the application provides access to the wagering-related transactions for the consumer, wherein the payload message is part of at least one of the wagering-related transactions and comprises at least one of: a request to place an online wager, or a response to the request to place the online wager.

37. (Canceled).

38. (Previously presented) The method of claim 15, further comprising:
providing an online gaming system by an OTLP host which uses the OLTP language, wherein the online gaming system processes wagering-related transactions; and
providing an application at a consumer location which uses the target program language, wherein the application provides access to the wagering-related transactions for the consumer,

wherein the payload message is part of at least one of the wagering-related transactions and comprises at least one of: a request to place an online wager, or a response to the request to place the online wager.

39. (Canceled).

39. (Previously presented) The method of claim 23, further comprising:

providing an online gaming system by an OTLP host which uses the OLTP language, wherein the online gaming system processes wagering-related transactions; and

providing an application at a consumer location which uses the target program language, wherein the application provides access to the wagering-related transactions for the consumer,

wherein the payload message is part of at least one of the wagering-related transactions and comprises at least one of: a request to place an online wager, or a response to the request to place the online wager.

40. (Canceled).

41. (Previously presented) The method of claim 28, further comprising:

providing an online gaming system by an OTLP host which uses the OLTP language, wherein the online gaming system processes wagering-related transactions; and

providing an application at a consumer location which uses the target program language, wherein the application provides access to the wagering-related transactions for the consumer,

wherein the payload message is part of at least one of the wagering-related transactions and comprises at least one of: a request to place an online wager, or a response to the request to place the online wager.

43. (Canceled).

44. (Previously presented) The machine readable medium of claim 32, wherein the set of stored instructions is capable of being executed by the processor to further:

provide an online gaming system by an OTLP host which uses the OLTP language, wherein the online gaming system processes wagering-related transactions; and

provide an application at a consumer location which uses the target program language, wherein the application provides access to the wagering-related transactions for the consumer, wherein the payload message is part of at least one of the wagering-related transactions and comprises at least one of: a request to place an online wager, or a response to the request to place the online wager.

45. (Canceled).

46. (Previously presented) The method of claim 1, wherein the program language is the target program language.

47. (Previously presented) The method of claim 15, wherein the program language is the target program language.

48. (Previously presented) The method of claim 23, wherein the program language is the target program language.

49. (Previously presented) The apparatus of claim 28, wherein the program language is the target program language.

50. (Previously presented) The machine readable medium of claim 32, wherein the program language is the target program language.

51. (Previously presented) The method of claim 1, wherein the automatically-generated source code is further configured to transform the first payload message directly from the target program language data format to the online transaction processing (OLTP) language message format, and to transform a second payload message directly from the OLTP language message format to the target program language data format.

52. (Previously presented) The method of claim 15, wherein the source code is further

configured to enable transformation of the payload message directly from the target program language data format to the OLTP language message format, and to enable transformation of a second payload message directly from the OLTP language message format to the target program language data format.

53. (Previously presented) The method of claim 23, wherein the source code is further configured, when compiled and executed, to transform the message directly from the target program language data format to the OLTP language message format, and to transform a second message directly from the OLTP language message format to the target program language data format.

54. (Previously presented) The apparatus of claim 28, wherein the source code is further configured to enable direct transformation of the message from the target program language message data format to the OLTP language message format, and to enable direct transformation of a second message from the OLTP language message format to the target program language message data format.

55. (Previously presented) The machine readable medium of claim 32, wherein the source code is further configured to transform the payload message directly from the target program language data format to the OLTP language message format, and to transform a second payload message directly from the OLTP language message format to the target program language data format.

56. (New) The method of claim 48, further comprising:
 using a stylesheet parser to compile the XML documents;
 receiving one or more name-value pairs for the message according to the program language; generating a byte array according to the OLTP language; and transmitting the byte array to an application;
 receiving a byte array for the message according to the OLTP language; generating one or more name-value pairs according to the program language; and transmitting the name-value pairs to an application; and
 wherein the object-oriented program language is a Java language.